



The staircasing effect in neighborhood filters and its solution

Antoni Buades, Bartomeu Coll, Jean-Michel Morel

► To cite this version:

Antoni Buades, Bartomeu Coll, Jean-Michel Morel. The staircasing effect in neighborhood filters and its solution. IEEE Transactions on Image Processing, 2006, 15 (6), pp.1499-1505. hal-00271143

HAL Id: hal-00271143

<https://hal.science/hal-00271143>

Submitted on 21 Jan 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

The staircasing effect in neighborhood filters and its solution

Antoni Buades, Bartomeu Coll, and Jean-Michel Morel

Abstract—Many classical image denoising methods are based on a local averaging of the color, which increases the signal/noise ratio. One of the most used algorithms is the neighborhood filter by Yaroslavsky or sigma filter by Lee, also called in a variant "SUSAN" by Smith and Brady or "Bilateral filter" by Tomasi and Manduchi. These filters replace the actual value of the color at a point by an average of all values of points which are simultaneously close in space and in color. Unfortunately, these filters show a "staircase effect", that is, the creation in the image of flat regions separated by artifact boundaries. In this paper, we first explain the staircase effect by finding the subjacent PDE of the filter. We show that this ill-posed PDE is a variant of another famous image processing model, the Perona-Malik equation, which suffers the same artifacts. As we prove, a simple variant of the neighborhood filter solves the problem. We find the subjacent stable PDE of this variant. Finally, we apply the same correction to the recently introduced NL-means algorithm which had the same staircase effect, for the same reason.

Edics: 2-NFLT Nonlinear Filtering and Enhancement. 2-REST Restoration.

I. INTRODUCTION

Many classical image denoising methods are based on a local average. The restored value at a pixel p is obtained as an average of its neighboring pixels. The most classical algorithm is the Gaussian filtering. In that case, the restored value is obtained as a weighted average where the weight at each pixel depends on the distance to the restored one. This low pass filter tends to blur the image.

The neighborhood filters avoid the blurring effect by restricting the average to pixels having a similar grey level value. The idea is that grey level values inside a homogeneous region slightly fluctuate while pixels belonging to different regions have a larger grey level difference. The Yaroslavsky neighborhood filter, [20], or sigma-filter [10], defines a neighborhood $G(\mathbf{x}, h) \cap B_\rho(\mathbf{x})$, where $B_\rho(\mathbf{x})$ is a ball of center \mathbf{x} and radius ρ and

$$G(\mathbf{x}, h) = \{\mathbf{y} \in \Omega \mid u(\mathbf{x}) - h < u(\mathbf{y}) < u(\mathbf{x}) + h\}.$$

Then, the filter takes an average of the values of pixels which are simultaneously close in grey level value and spatial distance. Many variants of the neighborhood filter have been introduced. Let us mention the SUSAN filter [17] and the bilateral filter [18] which are detailed in section II-A. These last filters have been further studied in [7], [3].

The strategy applied by neighborhood filters to avoid the blurring effect is similar to the one applied by some non linear PDE's. The early Perona-Malik "anisotropic diffusion" [13] reads

$$u_t = \operatorname{div}(g(|Du|^2)Du), \quad (1)$$

where $g : [0, +\infty) \rightarrow [0, +\infty)$ is a smooth decreasing function satisfying $g(0) = 1$, $\lim_{s \rightarrow +\infty} g(s) = 0$. The aim of this equation is

This work has been partially financed by the Centre National d'Etudes Spatiales (CNES), the Office of Naval Research under grant N00014-97-1-0839, the Ministerio de Ciencia y Tecnologia under grant TIC2002-02172. During this work, the first author had a fellowship of the Govern de les Illes Balears for the realization of his PhD.

A. Buades and B. Coll are with the University of Balearic Islands, Ctra. Valldemossa Km. 7.5, 07122 Palma de Mallorca, Spain (e-mail: toni.buades@cmla.ens-cachan.fr, tomeu.coll@uib.es)

J.M Morel is with the CMLA, ENS Cachan 61, Av du Prié 1/2 sident Wilson 94235 Cachan, France (e-mail: morel@cmla.ens-cachan.fr)

to apply a diffusion process inside the homogeneous regions, where $|Du|$ is small, while the diffusion is stopped near the boundaries or edges, where there is a large grey level difference between neighboring pixels and therefore a large gradient. The similarity between the neighborhood filters and the non linear PDE equations has been discovered in [3]. In the mentioned paper, which uses and extends anterior calculations in [15] and ideas of Weickert [19] it is actually proved that in one dimension, a discrete version of the Perona-Malik equation can be made similar to a 3x3 neighborhood filter.

In this paper we use a PDE formalism and, extending the results and technique of [3], prove that the neighborhood filter is equivalent to a Perona-Malik equation when the size of the spatial neighborhood tends to zero. The restored images by both methods are then compared. Mathematical and experimental evidences show that both filters share an undesirable shock effect that creates large flat zones and boundaries inside smooth regions.

We finally show that a simple modification of the neighborhood filter, a linear regression correction, allows us to avoid this shock effect and leads to more natural filtered images. The subjacent PDE of this new model explains the avoidance of the shock effect. This nonlinear PDE is equivalent to a heat equation when the image gradient is low and to the mean curvature motion when the gradient is large. Thus, it is an edge-preserving parabolic partial differential equation.

Our plan is as follows. In section II we give a brief introduction to neighborhood or bilateral filters and non-linear PDE's in image filtering. In section III we prove that neighborhood filters are asymptotically equivalent to the Perona-Malik equation as the size of the neighborhood tends to zero. In section IV we present a simple modification of the neighborhood filter, a linear regression correction, which avoids the shock effect and compute its subjacent PDE. Finally, in section V we show that the above linear correction can be applied to a recently introduced filter: The NL-means (Non local means) [4], [5], which is a strong improvement of neighborhood filters.

II. NEIGHBORHOOD FILTERS AND PDE'S

A. Neighborhood filters

We call neighborhood filter any filter which restores a pixel by taking an average of the values of neighboring pixels with a similar grey level value. Yaroslavsky (1985) [20] and Lee (1983) [10] averages pixels belonging to the neighborhood $G(\mathbf{x}, h) \cap B_\rho(\mathbf{x})$. This filter can be rewritten in a more continuous form as

$$YNF_{h,\rho}u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{B_\rho(\mathbf{x})} u(\mathbf{y}) e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} d\mathbf{y}, \quad (2)$$

where $\mathbf{x} \in \Omega$ and $C(\mathbf{x}) = \int_{B_\rho(\mathbf{x})} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} d\mathbf{y}$ is the normalization factor. The grey level threshold has been changed by a weighting function depending on a filtering parameter h . Only pixels inside $B_\rho(\mathbf{x})$ are averaged. This filter is less known than more recent versions, namely the *SUSAN filter* (1995) [17] and the *Bilateral filter* (1998) [18]. Both algorithms, instead of considering a fixed spatial neighborhood $B_\rho(\mathbf{x})$, weigh the distance to the reference pixel \mathbf{x} ,

$$SNF_{h,\rho}u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{\Omega} u(\mathbf{y}) e^{-\frac{|\mathbf{y}-\mathbf{x}|^2}{\rho^2}} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} d\mathbf{y}, \quad (3)$$

where $C(\mathbf{x}) = \int_{\Omega} e^{-\frac{|\mathbf{y}-\mathbf{x}|^2}{\rho^2}} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} d\mathbf{y}$ is the normalization factor and ρ is now a spatial filtering parameter.

In principle, there is no difference between $YNF_{h,\rho}$ and $SNF_{h,\rho}$. The performance of both algorithms is justified by the same arguments. Inside a homogeneous region, the grey level values slightly fluctuate because of the noise. In that case, the first strategy computes an arithmetic mean of the neighborhood and the second strategy a gaussian mean. At a contrasted edge separating two regions, if the grey level difference between both regions is larger than h , both algorithms compute averages of pixels belonging to the same region as the reference pixel. Thus, the algorithm does not blur the edges, which is its main scope. Unfortunately, it creates artificial shocks, which it is our aim to analyze.

B. Image filtering and enhancement by PDE models

Let us return to the Perona and Malik equation [13],

$$u_t = \text{div}(g(|Du|^2)Du), \quad (4)$$

where $g : [0, +\infty) \rightarrow [0, +\infty)$ is a smooth decreasing function satisfying $g(0) = 1$, $\lim_{s \rightarrow +\infty} g(s) = 0$. In order to interpret this equation, let us consider the second derivatives of u in the directions of Du and Du^\perp ,

$$u_{\eta\eta} = D^2u\left(\frac{Du}{|Du|}, \frac{Du}{|Du|}\right)(\mathbf{x}), \quad u_{\xi\xi} = D^2u\left(\frac{Du^\perp}{|Du|}, \frac{Du^\perp}{|Du|}\right)(\mathbf{x}),$$

where D^2u is the matrix of the second derivatives of u and all derivatives are evaluated at \mathbf{x} . Then, the equation (1) can be rewritten as

$$u_t = g(|Du|^2)u_{\xi\xi} + h(|Du|^2)u_{\eta\eta}, \quad (5)$$

where $h(s) = g(s) + 2sg'(s)$. This last equation is obtained by decomposing the divergence operator and taking into account that $|Du|^2 u_{\eta\eta}(\mathbf{x}) = u_x^2 u_{xx} + 2u_x u_y u_{xy} + u_y^2 u_{yy}$ and $\Delta u(\mathbf{x}) = u_{\xi\xi}(\mathbf{x}) + u_{\eta\eta}(\mathbf{x})$.

Perona and Malik proposed the function $g(s) = \frac{1}{1+s/k}$. In that case, the coefficient of the first term is always positive and therefore this term appears as a one dimensional diffusion in the orthogonal direction to the gradient. The sign of the second coefficient, however, depends on the value of the gradient. When $|Du|^2 < k$ this second term appears as a one dimensional diffusion in the gradient direction. It leads to a reverse heat equation term when $|Du|^2 > k$. The equation is ill-posed.

The Perona-Malik model has had many variants and extensions. Tannenbaum and Zucker [8] proposed, endowed in a more general shape analysis framework, the simplest equation of the list,

$$u_t = |Du| \text{div} \left(\frac{Du}{|Du|} \right) = u_{\xi\xi}.$$

This equation had been proposed some time before in another context by Sethian [16] as a tool for front propagation algorithms. It performs a “pure” diffusion in the direction orthogonal to the gradient and is related to two models proposed in image restoration. The Rudin-Osher-Fatemi [14] total variation model leads to minimize the total variation of the image $TV(u) = \int |Du|$, subject to some constraints. The steepest descent of this energy reads, at least formally,

$$\frac{\partial u}{\partial t} = \text{div} \left(\frac{Du}{|Du|} \right) \quad (6)$$

which is quite related to the mean curvature motion and to the Perona-Malik equation when $g(|Du|^2) = \frac{1}{|Du|}$. This particular case, which is not considered in [13], yields again (6). The existence of a solution and the qualitative properties of this curvature flow were studied in

[1], [2]. These authors study mathematically and actually demonstrate the existence of a staircase effect for this equation.

In the next section, we shall pile up a new argument in favor of this convergence of image restoration models by PDE towards variants of the curvature equation. We shall indeed prove that neighborhood filters are consistent with a Perona-Malik equation with an inverse diffusion term. We shall propose a straightforward improvement of the neighborhood filter. It consists of replacing the average by a linear regression. This improved neighborhood filter will be shown equivalent to a heat equation when the image gradient is low and to the mean curvature motion when the gradient is high.

III. RELATION BETWEEN THE NEIGHBORHOOD FILTERS AND THE PERONA-MALIK FILTER

In the next theorem we compute the asymptotic expansion of the Yaroslavsky neighborhood filter when $\rho, h \rightarrow 0$. We distinguish three different cases depending on the order of magnitude of h and ρ : h is much larger than ρ , both have the same order, or ρ is much larger than h . The first case takes us back to the heat equation and is therefore uninteresting. The second case leads to an equivalence between the Yaroslavsky neighborhood filter and the Perona-Malik equation. The third case is uninteresting again as it gives back a slightly anisotropic heat equation. Thus, one of the aims of the next theorem is also to give a relevant relationship between the space scale ρ and the grey level scale h .

Theorem 1: Suppose $u \in C^2(\Omega)$, and let $\rho, h, \alpha > 0$ such that $\rho, h \rightarrow 0$ and $h = O(\rho^\alpha)$. Let us consider the continuous function \tilde{g} defined by $\tilde{g}(t) = \frac{1}{3} \frac{te^{-t^2}}{E(t)}$, for $t \neq 0$, $\tilde{g}(0) = \frac{1}{6}$, where $E(t) = 2 \int_0^t e^{-s^2} ds$. Let \tilde{f} be the continuous function defined by

$$\tilde{f}(t) = 3\tilde{g}(t) + \frac{3\tilde{g}(t)}{t^2} - \frac{1}{2t^2}, \quad \tilde{f}(0) = \frac{1}{6}.$$

Then, for $\mathbf{x} \in \Omega$,

1) If $\alpha < 1$,

$$YNF_{h,\rho}u(\mathbf{x}) - u(\mathbf{x}) \simeq \frac{\Delta u(\mathbf{x})}{6} \rho^2.$$

2) If $\alpha = 1$,

$$YNF_{h,\rho}u(\mathbf{x}) - u(\mathbf{x}) \simeq \left[\tilde{g}\left(\frac{\rho}{h} |Du(\mathbf{x})|\right) u_{\xi\xi}(\mathbf{x}) + \tilde{f}\left(\frac{\rho}{h} |Du(\mathbf{x})|\right) u_{\eta\eta}(\mathbf{x}) \right] \rho^2$$

3) If $1 < \alpha < \frac{3}{2}$,

$$YNF_{h,\rho}u(\mathbf{x}) - u(\mathbf{x}) \simeq \tilde{g}(\rho^{1-\alpha} |Du(\mathbf{x})|) [u_{\xi\xi}(\mathbf{x}) + 3u_{\eta\eta}(\mathbf{x})] \rho^2.$$

Proof:

First, we rewrite the difference $YNF_{h,\rho}u(\mathbf{x}) - u(\mathbf{x})$ as

$$YNF_{h,\rho}u(\mathbf{x}) - u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{B_\rho(0)} (u(\mathbf{x}+\mathbf{t}) - u(\mathbf{x})) e^{-\frac{|u(\mathbf{x}+\mathbf{t})-u(\mathbf{x})|^2}{h^2}} d\mathbf{t},$$

and denote it by $d_{h,\rho}(\mathbf{x})$. We denote by $\eta = Du(\mathbf{x})/|Du(\mathbf{x})|$ and $\xi = Du(\mathbf{x})^\perp/|Du(\mathbf{x})|$ the gradient and tangent directions at \mathbf{x} . If $Du(\mathbf{x}) = 0$, we take an arbitrary pair of orthogonal unit vectors for η and ξ . Taking into account that ξ and η are orthogonal and with norm equal to 1, we use them to define a Cartesian reference frame centered at \mathbf{x} . Then, by taking the Taylor expansion of $u(\mathbf{x} + \mathbf{t})$ in the new coordinate system we obtain

$$u(\mathbf{x} + \mathbf{t}) - u(\mathbf{x}) = pt_1 + qt_1^2 + rt_2^2 + st_1t_2 + O(|\mathbf{t}|^3),$$

where $\mathbf{t} = (t_1, t_2)$, $p = |Du(\mathbf{x})|$ and if $p > 0$,

$$q = \frac{1}{2} D^2u\left(\frac{Du}{|Du|}, \frac{Du}{|Du|}\right)(\mathbf{x}), \quad r = \frac{1}{2} D^2u\left(\frac{Du^\perp}{|Du|}, \frac{Du^\perp}{|Du|}\right)(\mathbf{x}),$$

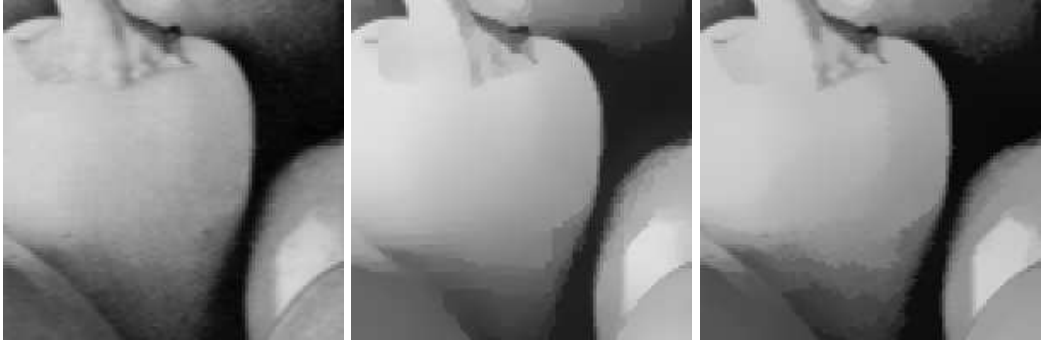


Fig. 1. Comparison between the neighborhood filter and the Perona-Malik equation. Left: original image. Middle: Perona-Malik filtered image. Right: filtered image by the neighborhood filter. The filtered images are very similar and both share artificial contours and flat zones that have been created inside the homogeneous regions.

$$s = D^2 u \left(\frac{Du^\perp}{|Du|}, \frac{Du}{|Du|} \right) (\mathbf{x}).$$

When $\alpha < 1$, we expand the exponential function and obtain

$$\begin{aligned} d_{h,\rho}(\mathbf{x}) &= \frac{1}{C(\mathbf{x})} \int_{B_\rho(0)} (u(\mathbf{x} + \mathbf{t}) - u(\mathbf{x})) \left(1 - \frac{p^2 t_1^2}{h^2} + O\left(\frac{|\mathbf{t}|^3}{h^2}\right) \right) \\ &\simeq \frac{1}{4\rho^2} \frac{2\Delta u(\mathbf{x})\rho^4}{3}. \end{aligned}$$

This proves (1). When $1 \leq \alpha < \frac{3}{2}$, we cannot apply the above expansion because $\frac{\rho^2}{h^2}$ does not tend to zero. However, $\frac{\rho^3}{h^2} \rightarrow 0$, and we can decompose the exponential as

$$e^{-\frac{|u(\mathbf{x}+\mathbf{t})-u(\mathbf{x})|^2}{h^2}} = e^{-\frac{p^2 t_1^2}{h^2}} \left(1 - \frac{2pt_1(qt_1^2 + rt_2^2 + st_1t_2)}{h^2} + O\left(\frac{|\mathbf{t}|^4}{h^2}\right) \right).$$

Using the Taylor expansion of u and of the exponential function we obtain

$$\begin{aligned} d_{h,\rho}(\mathbf{x}) &\simeq \frac{p}{2h\rho E(\frac{\rho}{h}p)} \left(4e^{-\frac{\rho^2}{h^2}p^2}\rho^4q + \frac{4e^{-\frac{\rho^2}{h^2}p^2}}{p^2}h^2\rho^2q \right. \\ &\quad \left. + \frac{4}{3}e^{-\frac{\rho^2}{h^2}p^2}\rho^4r - 2E(\frac{\rho}{h}p)h^3\rho \right). \end{aligned}$$

If $h \simeq \rho$, then all the terms of above expression have the same order ρ^2 and rewriting them proves (2). When $h \simeq \rho^\alpha$, $1 < \alpha < \frac{3}{2}$, we keep the term of lower order and get (3). ■

Interpretation: According to Theorem 1 the Yaroslavsky neighborhood filter acts as an evolution PDE with two terms. The first term is proportional to the second derivative of u in the direction ξ , which is tangent to the level line passing through \mathbf{x} . The second term is proportional to the second derivative of u in the direction η which is orthogonal to the level line passing through \mathbf{x} . The evolution equations $u_t = c_1 u_{\xi\xi}$ and $u_t = c_2 u_{\eta\eta}$ act as filtering or enhancing models depending on the signs of c_1 and c_2 . Following the previous theorem, we can distinguish three cases, depending on the values of h and ρ .

First, if h is much larger than ρ , both second derivatives are weighted by the same positive constant. In that case, the addition of both terms is equivalent to the Laplacian of u , Δu , and we get back to gaussian filtering.

Second, if h and ρ have the same order of magnitude, the neighborhood filter behaves as a filtering/enhancing algorithm. The weighting coefficient of the tangent diffusion, $u_{\xi\xi}$, is given by $\tilde{g}(\frac{\rho}{h}|Du|)$. The function \tilde{g} is positive and decreasing. Thus, there is always diffusion in that direction. The weight of the normal diffusion, $u_{\eta\eta}$, is given by $\tilde{f}(\frac{\rho}{h}|Du|)$. As the function \tilde{f} takes positive and negative values (see Figure 2), the filter behaves as a filtering/enhancing algorithm

in the normal direction and depending on $|Du|$. If \tilde{B} denotes the zero of \tilde{f} , then a filtering model is applied wherever $|Du| < \tilde{B}\frac{h}{\rho}$ and an enhancing strategy wherever $|Du| > \tilde{B}\frac{h}{\rho}$. The intensity of the filtering in the tangent diffusion and the enhancing in the normal diffusion tend to zero when the gradient tends to infinity. Thus, points with a very large gradient are not altered.

Finally, if ρ is much larger than h , the value $\frac{\rho}{h}$ tends to infinity and if the gradient of the image is bounded then the filtering magnitude $\tilde{g}(\frac{\rho}{h}|Du|)$ tends to zero. Thus, the original image is hardly altered.

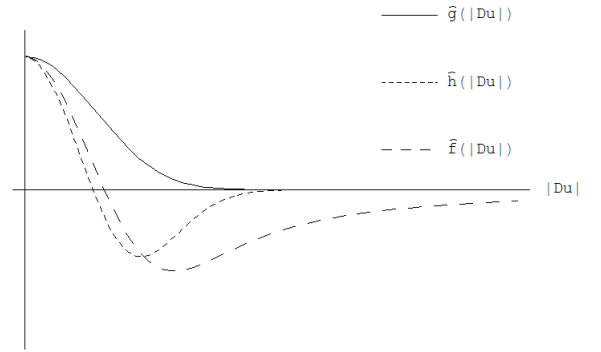


Fig. 2. Comparison of the neighborhood filter and the Perona-Malik filter. Magnitude of the tangent diffusion (continuous line) and normal diffusion (dashed line - -) of Theorem 1 in the case that $\rho = h$. Magnitude of the tangent diffusion (continuous line) and normal diffusion (dashed line - -) of the Perona-Malik model (7). Both models show nearly the same behavior.

We observe that when ρ and h have the same order, the neighborhood filter asymptotically behaves like a Perona-Malik model. Let us be more specific about this comparison. Taking $g(s) = \tilde{g}(s^{\frac{1}{2}})$ in the Perona-Malik equation (5), we obtain

$$u_t = \tilde{g}(|Du|)u_{\xi\xi} + \tilde{h}(|Du|)u_{\eta\eta}, \quad (7)$$

where $\tilde{h}(s) = \tilde{g}(s) + s\tilde{g}'(s)$. Thus, the Perona-Malik model and the neighborhood filter can be decomposed in the same way and with exactly the same weight in the tangent direction. Then the function \tilde{h} has the same behavior as \tilde{f} (Theorem 1), as can be observed in Figure 2. Thus in that case, a neighborhood filter has exactly the same qualitative behavior as a Perona-Malik model, even if we cannot rewrite it exactly as such.

Figure 1 displays an experiment comparing both methods. The filtered images are very similar, although the implementations are obviously very different. The neighborhood filter is implemented

exactly as in its definition and the Perona-Malik model by the explicit difference scheme proposed in the original paper. Both filters tend to create large flat zones and boundaries inside smooth regions as can be observed in the figure. Let us mention that similar calculations were performed in a particular case for the neighborhood median filter by Masnou [11].

IV. A REGRESSION CORRECTION OF THE NEIGHBORHOOD FILTER

In the previous section we have shown the enhancing character of the neighborhood filter. We have seen that the neighborhood filter, as the Perona-Malik model, can create large flat zones and spurious contours inside smooth regions. This effect depends upon a gradient threshold which is hard to fix in such a way as to always separate the visually smooth regions from edge regions. Thus, both models cannot be applied to images without some user's supervision. In particular, none of them gives satisfactory results in denoising software. In order to avoid this undesirable effect let us analyze it in more detail.

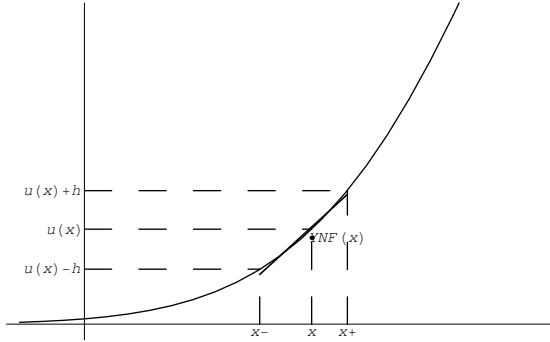


Fig. 4. Illustration of the shock effect of the YNF on a concave signal. The number of points y satisfying $u(x) - h < u(y) \leq u(x)$ is larger than the number satisfying $u(x) \leq u(y) < u(x) + h$. Thus, the average value $YNF(x)$ is smaller than $u(x)$, enhancing that part of the signal. The regression line of u inside $(x-, x+)$ better approximates the signal at x .

In Figure 4 we give a simple illustration of this effect with the Yaroslavsky neighborhood filter. For each x in the concave part of the signal, the filtered value is the average of the points y such that $u(x) - h < u(y) < u(x) + h$ for a certain threshold h . As it is illustrated in the figure, the number of points satisfying $u(x) - h < u(y) \leq u(x)$ is larger than the number of points satisfying $u(x) \leq u(y) < u(x) + h$. Thus, the average value $YNF(x)$ is smaller than $u(x)$ and the concavity of the signal is enhanced. A similar argument can be applied in the convex parts of the signal, dealing to the opposite enhancing effect. Therefore, shocks appear at inflexion points, where concave and convex parts meet. Figure 4 also shows that the mean is not a good estimate of $u(x)$ in that case. In the same figure, we display the regression line approximating u inside $(u^{-1}(u(x) - h), u^{-1}(u(x) + h))$. The value of the regression line at x better approximates the signal. In the sequel, we propose to correct the neighborhood filter with this better estimate.

Definition 1 (Linear regression neighborhood filter): We call $LYNF_{h,\rho}u(\mathbf{x})$ the value obtained at $\mathbf{x} = (x_1, x_2)$ by finding the plane locally approximating u in the following sense,

$$\min_{a,b,c} \int_{B_\rho(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) (u(\mathbf{y}) - ay_1 - by_2 - c)^2 d\mathbf{y} \quad (8)$$

where

$$w(\mathbf{x}, \mathbf{y}) = e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}}.$$

Then, the restored value at \mathbf{x} is given by $ax_1 + bx_2 + c$. The weights used to define the minimization problem are the same as the ones used by the neighborhood filter. Thus, the points with a grey level

value close to $u(\mathbf{x})$ will have a stronger influence in the minimization process. The only difference with YNF is the replacement of an average by a linear regression. The minimization process is made explicit, since we can easily derive the normal equations. Thus, the computation of the above linear regression reduces to the solution of a 3×3 linear system.

Next, we compute the asymptotic behavior of the filter when ρ and h have the same order and tend to zero. This has shown to be the interesting case in Theorem 1, but also the one causing a shock effect.

Theorem 2: Suppose $u \in C^2(\Omega)$, and let $\rho, h > 0$ such that $\rho, h \rightarrow 0$ and $O(\rho) = O(h)$. Let \tilde{h} be the continuous function defined by $\tilde{h}(0) = \frac{1}{6}$,

$$\tilde{h}(t) = \frac{E(t) - 2te^{-t^2}}{2t^2 E(t)}$$

for $t \neq 0$, where $E(t) = 2 \int_0^t e^{-s^2} ds$. Then $\tilde{h}(t)$ is positive decreasing and

$$LYNF_{h,\rho}u(\mathbf{x}) - u(\mathbf{x}) \simeq \left[\frac{1}{6} u_{\xi\xi}(\mathbf{x}) + \tilde{h}\left(\frac{\rho}{h}\right) |Du(\mathbf{x})| u_{\eta\eta}(\mathbf{x}) \right] \rho^2.$$

Proof: We can suppose without loss of generality that $\mathbf{x} = (0, 0)$. In that case, $LYNF_{h,\rho}u(0, 0) = c$, where c is given by the solution of (8). By straightforward computations, it is easy to prove that $c = c_1/c_2$, where

$$c_1 = c_{020}(c_{200}c_{001} - c_{100}c_{101}) + c_{110}(c_{100}c_{011} - c_{110}c_{001}) + c_{010}(c_{110}c_{101} - c_{200}c_{011})$$

$$c_2 = c_{020}(c_{200}c_{000} - c_{100}^2) + c_{110}(c_{010}c_{100} - c_{110}c_{000}) + c_{010}(c_{100}c_{110} - c_{010}c_{200})$$

$$\text{and } c_{ijm} = \int_{B_\rho(0,0)} w(t_1, t_2) t_1^i t_2^j u(t_1, t_2)^m dt_1 dt_2.$$

By the same arguments of Theorem 1 we take the following Taylor expansion of $u(\mathbf{t})$

$$u(\mathbf{t}) = u(0, 0) + pt_1 + qt_1^2 + rt_2^2 + st_1t_2 + O(|\mathbf{t}|^3)$$

and

$$w(\mathbf{t}) = e^{-\frac{p^2 t_1^2}{h^2}} \left(1 - \frac{2pt_1(qt_1^2 + rt_2^2 + st_1t_2)}{h^2} + O\left(\frac{|t|^4}{h^2}\right) \right),$$

where the differential operators p, q, r, s are defined in the proof of Theorem 1 and are evaluated at $(0, 0)$. Taking these approximations of u and w into c , we obtain

$$c - u(0, 0) \simeq \frac{8e^{-2d^2 p^2} (dp)^2 - 8e^{-d^2 p^2} E(dp) dp + 2E(dp)^2}{(dp)^2 (2E(dp)^2 - 4e^{-d^2 p^2} E(dp)(dp))} q + \frac{1}{3} r,$$

where $d = \rho/h$. The result of the theorem follows. \blacksquare

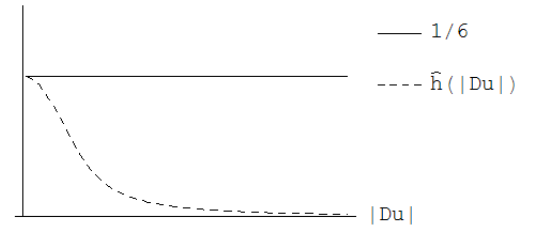


Fig. 5. Weight functions of Theorem 2. Constant function $1/6$ (continuous line) and function \tilde{h} (dashed line).

According to the previous theorem, the filter can be written as the sum of two diffusion terms in the direction of ξ and η . The behavior of the weight functions is quite different from the neighborhood filter case. The function weighting the tangent diffusion is a positive constant. The function weighting the normal diffusion is positive

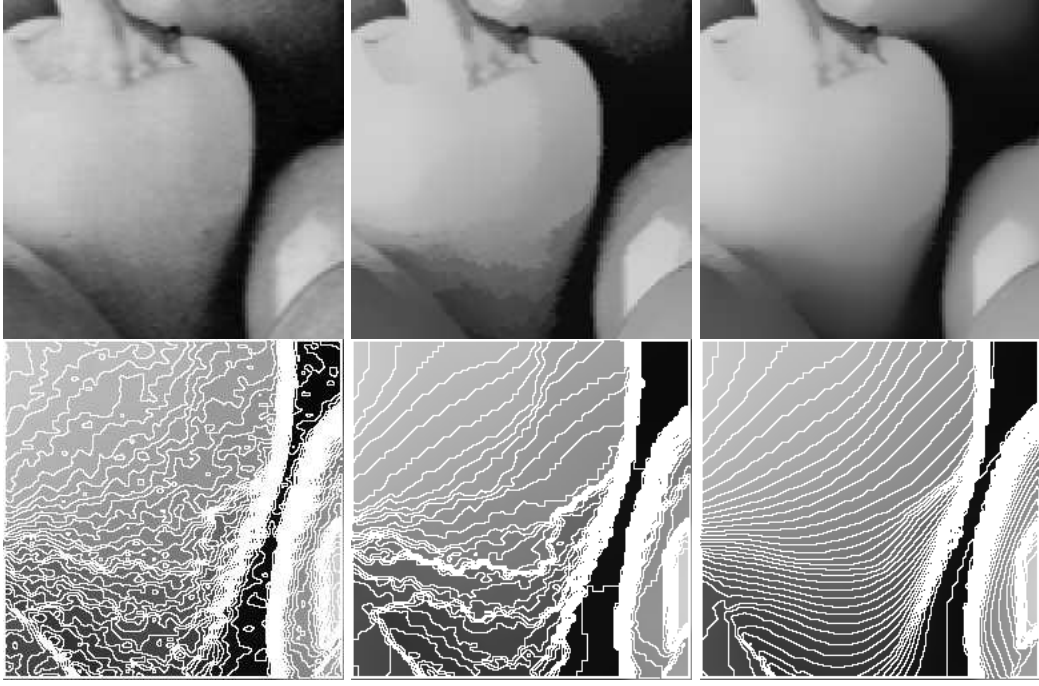


Fig. 3. Comparison of the neighborhood filter and the linear regression correction. Top left: original image. Top middle: filtered image by the neighborhood filter. Top right: filtered image by the regression neighborhood filter. Bottom: level lines of a part of the images on the above line. Both neighborhood filters have been performed with the same filtering parameters and the same number of iterations. The linear regression neighborhood algorithm has filtered the image while preserving the main boundaries as the original neighborhood filter. No enhancing has been applied by the linear correction avoiding the shock effect. The level lines of the neighborhood filter tend to group and create large flat zones. In addition, these level lines oscillate while those of the linear regression algorithm have been correctly filtered.

and decreasing, and therefore there is no enhancement effect. The algorithm combines the tangent and normal diffusions when the gradient is small. When the gradient is large, the normal diffusion is cancelled and the image is filtered only in its tangent direction. This diffusion makes the level lines evolve proportionally to their curvature. In the Perona-Malik model the diffusion is stopped near the edges. In this case, the edges are filtered by a mean curvature motion.

Figure 3 displays an experiment comparing the neighborhood filter and the linear regression correction. The linear regression neighborhood algorithm has filtered the image and preserved the main boundaries, as the original neighborhoods filters do. The figure shows that no artificial enhancement has been applied by the linear correction, thus avoiding the shock effect. Figure 3 also displays the level lines of the filtered images which corroborate the above observations. The level lines of the neighborhood filter tend to group and create large flat zones. In addition, these level lines oscillate while those of the linear regression correction have been correctly filtered.

V. THE NL-MEANS AND THE REGRESSION CORRECTION

The above regression correction applied to the neighborhood filter can be applied to all filters involving a local average. Such filters are characterized by a family of weights $w(\mathbf{x}, \mathbf{y})$ for $\mathbf{y} \in B_\rho(\mathbf{x})$ which can be modified by the previous argument. The weight $w(\mathbf{x}, \mathbf{y})$ reflects the influence that pixel \mathbf{y} has in the minimization process. Let us apply this strategy to the filter recently introduced in [5], [6]. In that case, the similarity between the pixels \mathbf{x} and \mathbf{y} is measured by the grey level differences in a whole Gaussian neighborhood of \mathbf{x} and \mathbf{y} . The NL-means (Non Local Means) algorithm is defined by

$$NL_h[u](\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{\Omega} w(\mathbf{x}, \mathbf{y}) u(\mathbf{y}) d\mathbf{y}, \quad (9)$$

where

$$w(\mathbf{x}, \mathbf{y}) = e^{-\frac{1}{h^2} \int_{\mathbb{R}^2} G_a(t) |u(\mathbf{x}+t) - u(\mathbf{y}+t)|^2 dt},$$

$C(\mathbf{x}) = \int_{\Omega} w(\mathbf{x}, \mathbf{y}) d\mathbf{y}$, G_a is a Gaussian kernel of standard deviation a and h acts as a filtering parameter. A recent paper by Kindermann et al. [9] gives a variational interpretation of the neighborhood and NL-means filters.

For pixels \mathbf{x} and \mathbf{y} , the NL-means algorithm does not only compare the grey level values $u(\mathbf{x})$ and $u(\mathbf{y})$, but also the grey level values in a Gaussian neighborhood. This permits a more robust comparison based on the detailed configurations of the neighborhoods of \mathbf{x} and \mathbf{y} . For example, a pixel \mathbf{y} such that $u(\mathbf{x}) = u(\mathbf{y})$ can have a very small or nearly zero weight in the restoration of \mathbf{x} , since the configurations around \mathbf{x} and \mathbf{y} can be very different. For a more exhaustive description of the NL-means and the comparison with the neighborhood filters and other algorithms see [4].

In order to apply the regression correction to the NL-means algorithm it is sound to restrict the search zone for a pixel $\mathbf{x} = (x_1, x_2)$ to a neighborhood $B_\rho(\mathbf{x})$. The filtered value is given by $ax_1 + bx_2 + c$, where a, b, c minimize

$$\min_{a,b,c} \int_{B_\rho(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) (u(\mathbf{y}) - ay_1 - by_2 - c)^2 d\mathbf{y}$$

and

$$w(\mathbf{x}, \mathbf{y}) = e^{-\frac{1}{h^2} \int_{\mathbb{R}^2} G_a(t) |u(\mathbf{x}+t) - u(\mathbf{y}+t)|^2 dt}.$$

Figure 6 displays a denoising experiment with the NL-means algorithm. The filtered image presents a shock effect similar to the one of the neighborhood filter. The regression correction avoids these shocks and restores a more natural image. Finally, Figures 7 compares the visual quality of the filtered images by the linear corrections of the neighborhood filter and the NL-means.

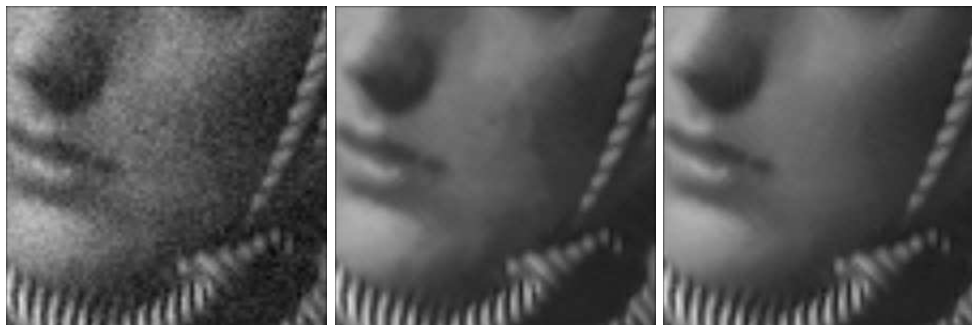


Fig. 6. Comparison of the NL-means and its regression correction on a noisy image. Left: Noisy image. Middle: NL-means filtered image. Right: Regression correction image. The face filtered by NL-means presents a shock effect similar to the neighborhood filter. These shocks are not created by the corrected filter leading to a more natural image.



Fig. 7. Comparison of the linear regression neighborhood filter and the linear regression NL-means. Left: Noisy image. Middle: Linear regression neighborhood filtered image. Right: Linear regression NL-means filtered image.

REFERENCES

- [1] F. Andreu, C. Ballester, V. Caselles and J.M. Mazón, "Minimizing Total Variation Flow", *Differential and Integral equations*, vol. 14(3), pp. 321-360, 2001.
- [2] F. Andreu, V. Caselles, J.I. Diaz and J.M. Mazón, "Qualitative Properties of the Total Variation Flow", *Journal of Functional Analysis* 188 (2), pp. 516-547, 2002.
- [3] D. Barash "A Fundamental Relationship between Bilateral Filtering, Adaptive Smoothing and the Nonlinear Diffusion Equation" *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (6), 2002.
- [4] A. Buades, B. Coll and J.M. Morel, "A review of image denoising methods, with a new one", accepted in *Multiscale Modeling and Simulation (SIAM)*. Preprint available at <http://www.cmla.ens-cachan.fr/Cmla/Publications/2004>.
- [5] A. Buades, B. Coll and J.M. Morel, "A non-local algorithm for image denoising", *IEEE International Conference on Computer Vision and Pattern Recognition*, 2005.
- [6] A. Buades, B. Coll and J.-M. Morel, "Image data process by image noise reduction and camera integrating the means for implementing this process", French patent pending 0404837, May 5, 2004.
- [7] M. Elad, "On the origin of the bilateral filter and ways to improve it", *Transactions on Image Processing*, 11 (10), pp. 1141-1151, 2002.
- [8] B.B. Kimia, A. Tannenbaum, and S.W. Zucker, "On the evolution of curves via a function of curvature I the classical case", *Journal of Mathematical Analysis and Applications*, 163(2), pp. 438-458, 1992.
- [9] S. Kindermann, S. Osher and P. Jones, "Deblurring and Denoising of Images by Nonlocal Functionals", *UCLA Computational and Applied Mathematics Reports*, 04-75, 2004.
- [10] J.S. Lee, "Digital image smoothing and the sigma filter", *Computer Vision, Graphics, and Image Processing*, 24, pp. 255-269, 1983.
- [11] S. Masnou, "Filtrage et désocclusion d'images par méthodes d'ensembles de niveau", PhD dissertation, Université Paris-IX Dauphine, 1998.
- [12] S. Osher and L. Rudin, "Feature oriented image enhancement using shock filters" *SIAM J. Numerical Analysis*, 27, pp. 919-940. 1990.
- [13] P. Perona and J. Malik, "Scale space and edge detection using anisotropic diffusion," *IEEE Trans. Patt. Anal. Mach. Intell.*, 12, pp. 629-639, 1990.
- [14] L. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms", *Physica D*, 60, pp. 259-268, 1992.
- [15] P. Saint-Marc, J.S. Chen, and G. Medioni, "Adaptive Smoothing: A General Tool for Early Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence* vol. 13 (6), pp. 514, 1991.
- [16] J. Sethian "Curvature and the evolution of fronts", *Comm. Math. Phys.*, 101, 1985.
- [17] S.M. Smith and J.M. Brady, "SUSAN - a new approach to low level image processing," *International Journal of Computer Vision*, Volume 23 (1), pp. 45-78, 1997.
- [18] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *Sixth International Conference on Computer Vision*, pp. 839-46. 1998.
- [19] J. Weickert, "Anisotropic Diffusion in Image Processing". Tuebner Stuttgart, 1998.
- [20] L.P. Yaroslavsky, *Digital Picture Processing - An Introduction*, Springer Verlag, 1985.